Towards a Distributed Congestion Control mechanism for Smart Grid Neighborhood Area Networks

Juan Pablo Astudillo Leon, Thomas Begin, Anthony Busson, Luis Javier de La Cruz Llopis

Towards a Distributed Congestion Control mechanism for Smart Grid Neighborhood Area Networks

JUAN PABLO ASTUDILLO LEÓN, Universitat Politècnica de Catalunya (UPC)
Department of Network Engineering
THOMAS BEGIN, Univ Lyon, Université Claude Bernard Lyon 1,
ENS de Lyon, Inria, CNRS
ANTHONY BUSSON, Univ Lyon, Université Claude Bernard Lyon 1,
ENS de Lyon, Inria, CNRS
LUIS J. DE LA CRUZ LLOPIS, Universitat Politècnica de Catalunya (UPC)
Department of Network Engineering

The need for significant improvements in the management and efficient use of electrical energy has led to the evolution from the traditional electrical infrastructures towards modern Smart Grid networks. Taking into account the critical importance of this type of networks, multiple research groups focus their work on issues related to the generation, transport and consumption of electrical energy. One of the key research points is the data communication network associated with the electricity transport infrastructure, and specifically the network that interconnects the devices in consumers' homes, the so-called Neighborhood Area Networks (NANs). In this paper, a new distributed congestion control mechanism is proposed, implemented and evaluated for NANs. Besides, different priorities have been considered for the traffic flows transmitted by different applications. The main goal is to provide with the needed Quality of Service (QoS) to all traffic flows, especially in high traffic load situations. The proposal is evaluated in the context of a wireless ad hoc network made up by a set of smart meter devices, using the Ad hoc On-Demand Distance Vector (AODV) routing protocol and the IEEE 802.11ac physical layer standard. The application of the proposed congestion control mechanism, together with the necessary modifications made to the AODV protocol, lead to performance improvements in terms of packet delivery ratio, network throughput and transit time, fairness between different traffic sources and QoS provision.

CCS Concepts: • **Networks → Traffic engineering algorithms**; *Network performance evaluation*; *Network measurement*;

Additional Key Words and Phrases: Smart Grid, Neighborhood Area Networks, Congestion Control

ACM Reference Format:
Juan Pablo Astudillo León, Thomas Begin, Anthony Busson, and Luis J. de la Cruz Llopis. 2019. Towards a Distributed Congestion Control mechanism for Smart Grid Neighborhood Area Networks. 1, 1 (November 2019), 17 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

Electricity is an essential resource widely used in our daily activities. Taking into account the importance of this resource, research groups have focused their efforts to improve the management and operation of the traditional electricity distribution networks. To this end, in Smart Grids a data communication network has been incorporated

Authors' addresses: Juan Pablo Astudillo León, Universitat Politècnica de Catalunya (UPC)
Department of Network Engineering, Barcelona, Spain, juan.pablo.astudillo@upc.edu; Thomas Begin, Univ Lyon, Université Claude Bernard Lyon 1,
ENS de Lyon, Inria, CNRS, Lyon, France, thomas.begin@univ-lyon1.fr; Anthony Busson, Univ Lyon, Université Claude Bernard Lyon 1,
ENS de Lyon, Inria, CNRS, Lyon, France, anthony.busson@univ-lyon1.fr; Luis J. de la Cruz Llopis, Universitat Politècnica de Catalunya (UPC)
Department of Network Engineering, Barcelona, Spain, luis.delacruz@upc.edu.

2019. Manuscript submitted to ACM

into the electric infrastructure. The objective is to provide new services to both the supplying companies and to their costumers [14]. In a society where new technologies are growing fast, a big amount of new services can be offered by the Smart Grid data communication network. Generally speaking, most Smart Grid applications have strong security and reliability requirements. Therefore, keeping the desired Quality of Service (QoS) for both classical and emerged applications is of paramount importance when planning and operating this type of networks [7].

The Smart Grid data communication network is divided into three different sub-networks: Home Area Network (HAN), Neighborhood Area Network (NAN) and Wide Area Network (WAN) [12]. On the one hand, the HAN interconnects the smart meters (SM) and other possible appliances present inside the homes. Some standards for HANs are IEEE 802.15.4 (Low-Rate Wireless Personal Area Networks, LR-WPAN) or the traditional IEEE 802.11 (Wireless Local Area Networks, WLAN). In turn, HANs are interconnected through the NANs. Here, selectable technologies are the well-known Power Line Communication (PLC), or wireless standards such as IEEE 802.15.4g, IEEE 802.11s and also WLAN. Finally, WAN allows the information exchange between the different NANs and the control center. A wired backbone or a wireless technology such as Wireless Metropolitan Area Networks (WirelessMAN, IEEE 802.16) can be considered.

The goal of this paper is to present and evaluate a distributed congestion control mechanism that allows improving the performance offered by the NANs, when the selected technologies are IEEE 802.11 Wireless Ad Hoc Network (WANET) and IEEE 802.11ac physical layer standard. Our solution is conceptually simple as well as easily tunable and implementable. In addition, it is independent of the routing protocol, mac and physical layers. The proposed mechanism combines several algorithms that allow differentiating the quality of service offered to each traffic based on its criticality, while providing a fair service to all nodes in the network. Note that our proposed solution can be viewed as a middleware candidate for Smart Grids that will perform over the network infrastructure with the aim of ensuring a fair sharing of the resources between nodes and classes as well as high levels for resource utilization and packet delivery ratio.

The rest of the paper is organized as follows. Section 2 discusses related work. The proposed solution is presented in Section 3, and its performance is evaluated in Section 4. Finally, Section 5 concludes the paper.

2 RELATED WORK

Several researchers have focused their efforts with the aim of improving the performance offered by the smart grid neighborhood area networks. For instance, authors in [16] present a performance evaluation and comparison of Optimized Link State Protocol (OLSR) [2] and HWMP (IEEE 802.11s) [5] routing protocols, where a classification of the main AMI (Advanced Metering Infrastructure) application traffics is taking into account. Later, in [15], the same authors propose an enhancement of the OLSR protocol by using the combination of different basic metrics. They use Relevant Link Metric Types (RLMTs) for each application, together with the AHP (Analytical Hierarchy Process) algorithm in order to select the best path. Based on IEEE 802.11s, a multigate communication network is proposed in [6] for improving the network performance for Smart Grids. They present a multi gateway, multichannel and a real-time traffic scheduling. On the other hand, to ensure the needed QoS of different Smart Grid applications, in [4] a modification to the HWMP protocol is presented (HWMP-NQ). They also present a modification of the airtime link metric by considering the packet size and the transmission rate. Finally, they present the benefits of a multi-gateway backup routing, and a modification of the path error mechanism to reduce the routing overhead.

Given that most data traffic is transmitted upstream from the smart meters to the data concentrator, authors in [9] present modifications to the airtime link metric calculation and to the path selection mechanism, in order to provide greater relevance to the upstream transmissions. Besides, authors address the need for congestion control when the network size is increased. Later, some of the authors of [9] analyze in [10] the reliability and the weaknesses of the

HWMP protocol when it is integrated to Smart Grid networks. Authors propose some solutions based again on some modifications to the airtime link metric and in the possibility of routes reservation. Besides, they present a novel delay-tolerant traffic management method in order to provide a better quality of service to some applications.

A different modification of the HWMP protocol was presented in [1], to allow an efficient selection of paths among multiple possibilities, depending on the service quality needs of the different traffic flows. Also, the mechanism is complemented with the assignment of different frequency channels to each available path. The main goal is to transmit priority traffics through the best paths, and using also the less congested channel. Besides, different channels are used for data and control traffic. Later, in [11], a multi-channel allocation scheme together with a congestion control mechanism was presented and evaluated, considering the different service quality needs of the different applications. The congestion control mechanism takes into account if the network is in an emergency state caused by intrinsic or extrinsic reasons. Different congestion control functions are proposed for each traffic, which provide a higher transmission probability to the traffic with higher QoS needs, based on the network congestion level and on the emergency state.

In this work, a different solution to improve the network performance for NANs is proposed, implemented and evaluated. Instead of considering IEEE 802.11s mesh networks, the basic IEEE 802.11 adhoc networks and the recent IEEE 802.11ac physical layer standard have been chosen. On the other hand, unlike the strategy followed in [1] and [11] where relay nodes are in charge of discarding packets in network congestion situations, in this paper it is up to the source nodes to reduce their packet generation rate. Thus, depending on the sensed degree of network congestion, source nodes may increase or decrease the rate at which they generate their packets. This way, we avoid unnecessary transmissions of packets that will be likely discarded later on their way to their destination. To this end, new signaling messages must be transmitted from relay nodes to source nodes. Besides, the proposed mechanism is also designed to provide a fair distribution of the available network resources between all the source nodes, avoiding a higher utilization by the nodes with higher packet generation rate or simply favorably located. Finally, our solution is independent of the routing protocol used.

3 PROPOSED SOLUTION

Before fully detailing our proposed mechanism for congestion control, we discuss some of its properties. The algorithms that compose the proposed mechanism are executed individually in every node (be it a source, relay or both). The amount of signaling traffic due to our mechanism is typically relatively small compared to the network traffic intensity, and overall the total traffic load is expected to be significantly reduced since our solution throttles the source nodes. In addition, the proposed algorithms do not significantly load the CPU of the nodes, which is advisable if we take into account that in many cases these devices are built on a large scale and at low cost, and so they could have limited resources.

Next subsections describe in detail the system and the designed algorithms that build the congestion control mechanism. In order to have a quick reference for all the variables used in the algorithms, Table 1 provides their definition and description.

3.1 Systems under study

Figure 1 shows the main NAN scenario used to evaluate our distributed solution. This NAN scenario is made up of a set of smart devices arranged in a grid topology, where data packets travel from the devices to the data concentrator and vice versa. Home users (smart meter devices and other utilities present at the HANs) transmit data to the Smart Grid control center through the data concentrator. Examples of data are meter reading data, home energy and Electric

Table 1. Definition of the variables.

Parameter	Parameters shared by all nodes Description	Units
K	number of classes	
N	number of nodes	
S	number of sources	
B	upper threshold on ρ, $B \in [0, 1]$	
C	lower threshold on ρ, $C \in [0, 1]$	
A	period at which the different algorithms are implemented	sec
I	percentage of traffic to increase, $I \geq 1$	
D	percentage of traffic to decrease, $D \in [0, 1]$	
δ^k	increment of each class k	bps
α^k	weight of each class k	
Parameter	**Local variables (sources or relay nodes)** **Description**	**Units**
ρ_n	current channel utilization factor at node n $\rho_n \in [0,1]$	
x_j^k	application rate demanded by node j of class k	bps
y_j^k	application rate from node j of class k	bps
$z_{j,n}^k$	constraint computed by node n for flows of class k from node j	bps
s_j^k	current tolerated application rate by node j of class k	bps
t_j^k	attained threshold by the application by node j of class k	bps

Vehicle (EV) charging information, among others. On the other hand, Power Quality (PQ) data is constantly measured, and depending on the situation, the data concentrator may transmit downstream Demand Response information to home users to adjust, for instance, energy consumption during peak hours.

In our proposal, applications are grouped into four different classes depending on their relevance. As mentioned, one of the characteristics of the congestion control mechanism is that it will favor the transmission of packets belonging to high classes, but at the same time providing sufficient transmission opportunities to low categories. Said differently, absolute priority is not given to high categories over low ones. In this way, it is intended to offer all categories their minimum service quality requirements in terms of throughput and network transit time. Note that this behavior is different from the one presented by the mechanisms that assign absolute priority to some traffic classes over others, such as the EDCA medium access control. Absolute prioritization is typically convenient when priority traffic represents a small fraction of the total, which is not necessarily true here. Besides, EDCA is not a network congestion control mechanism, and thus buffers can indiscriminately overflow in the absence of a mechanism that explicitly deals with congestion, as our proposed solution does.

3.2 Distributed congestion control mechanism

To build the congestion control mechanism, three different algorithms, together with new signaling messages, have been designed, implemented and evaluated. For the sake of clarity, a first and simple scenario (Figure 2) will be considered to

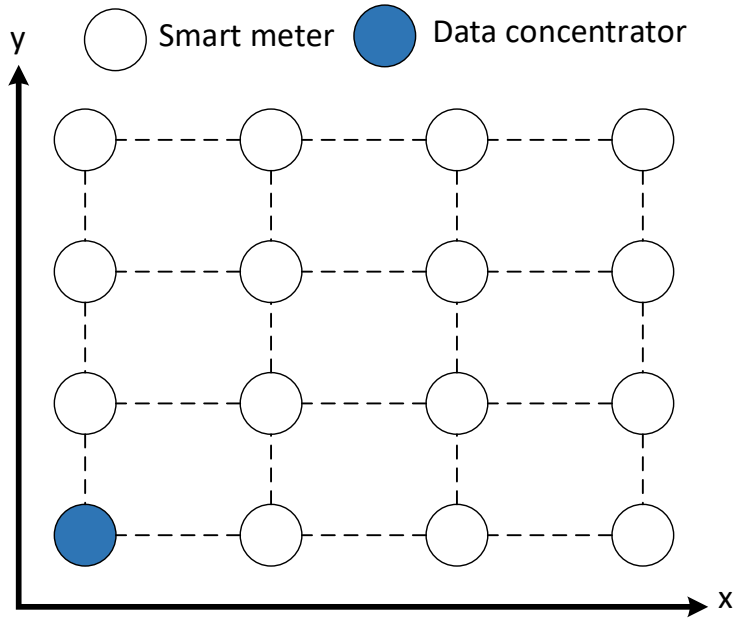

Fig. 1. Smart Grid Neighborhood Area Networks scenario

detail the algorithms. This scenario will also be used in next section, together with the more realistic one (Figure 1), to carry out the performance evaluation of the proposal.

In the first scenario, source nodes (S1, S2 and S3) transmit different NAN applications to the destination node (D). Relay nodes (R1 and R2) are in charge of the retransmission of the data packets. It must be kept in mind that relay nodes play a central role in our solution. The objective is to avoid saturation in the relay nodes by adjusting the current application rates from the source nodes. To ensure traffic differentiation to the NAN applications with different QoS needs, different algorithm parameters can be tuned.

3.2.1 Relay nodes algorithm to detect network congestions. The distributed congestion mechanism is based on the value of the current channel utilization factor. This parameter is measured by each device n and it is denoted as ρ_n. It represents the proportion of time where the channel is sensed as busy. It is evaluated as the average value over a period of time A.

Algorithm 1 is performed independently on all relay nodes. This algorithm has two main tasks. On the one hand, each node n updates its current value for the utilization factor ρ_n and estimates the amount of traffic that passes through them for each class and each source node. We use y_j^k to denote the measured traffic rate of class k originated at source j. On the other hand, Algorithm 1 is also in charge of notifying the source nodes so that they regulate their packet generation rates whenever a network congestion is detected. In practice, when the current node becomes saturated ($\rho_n \geq B$), it requests the sources to reduce their packet generation rate by a D percentage. Otherwise, the node lets the

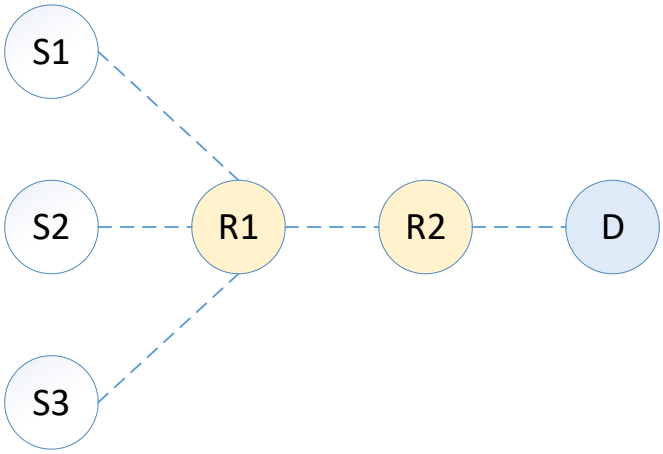

Fig. 2. Tree topology scenario

source nodes increase their rate by an I percentage. For this purpose, a new signaling packet has been defined, which is either sent or not to the source nodes based on the current value of ρ_n.

Algorithm 1: Run at each relay node n to detect network congestions.

Input: Parameters A seconds, B, C

1 **while** *true* **do**
2 update value of ρ_n and y_j^k
3 **if** $\rho_n \geq B_n$ **then**
4 control flows: compute smaller values of y_j^k following Algorithm 2 (y_j^k, D)
5 notify the sources with new values of $z_{j,n}^k$
6 **else if** $\rho_n \leq C_n$ **then**
7 relax flows, authorize larger values of y_j^k following Algorithm 2 (y_j^k, I)
8 notify the sources with the new values of $z_{j,n}^k$
9 wait A seconds

3.2.2 Relay nodes algorithm to determine the constrained packet generation rates for source nodes. Algorithm 2 is also implemented on the relay nodes. It computes the packet generation rates that will be notified to the sources depending on the different network congestion situations. This algorithm may be called to increase (with proportion I) or decrease (with proportion D) the total transmission rate from the sources. More precisely, the amount of bandwidth that will be allocated is $F \cdot \sum y_j^k$ (with $F = I$ or $F = D$) and where $\sum y_j^k$ represents the overall transmission rate measured by the relay node. The transmission rate that will be assigned to a flow from source j with priority k is initially set to 0 ($z_{j,n}^k = 0$). In a first time (lines $4-7$), these transmission rates are increased in a while loop with increments equal to δ_k. For a given flow, it stops increasing when a proportion α_k of the current rate is reached ($z_{j,n}^k < t_j^k$). Both thresholds α_k and δ_k depend on the flows priority k with typically $\alpha^1 > \alpha^2 > \cdots > \alpha^K$ and $\delta^1 > \delta^2 > \cdots > \delta^K$. This first step aims to provide a fair proportion ($\alpha_.$) to each flow according to their current throughput and priority. In a second step (lines

8 − 11), if the threshold on the overall transmission rate has not been reached ($\sum z_{j,n}^k < F \cdot \sum y_j^k$), the resources left is shared among the class and sources with the same principle, i.e. increasing with increment δ_k.

Algorithm 2: Run at each relay node n to determine the constrained packet generation rates for source nodes.

Input: Parameters y_j^k, F = I or D (percentage increase or decrease of the source rates)
Output: $z_{j,n}^k$

δ^k : defines the increments for each class
$t_j^k = \alpha_k \cdot y_j^k$: calculate the thresholds for each source and class
$z_{j,n}^k = 0$
while $\sum_{j,k} z_{j,n}^k < F \cdot \sum_{j,k} y_j^k$ **do**
  for (j, k) *in* $\{1, ..., S\} \times \{1, ..., K\}$ **do**
    if $z_{j,n}^k < t_j^k$ **and** $\sum_{j,k} z_{j,n}^k < F \cdot \sum y_j^k$ **then**
      $z_{j,n}^k = z_{j,n}^k + \delta^k$
while $\sum_{j,k} z_{j,n}^k < F \cdot \sum_{j,k} y_j^k$ **do**
  for (j, k) *in* $\{1, ..., S\} \times \{1, ..., K\}$ **do**
    if $\sum_{j,k} z_{j,n}^k < F \cdot \sum y_j^k$ **then**
      $z_{j,n}^k = z_{j,n}^k + \delta^k$

3.2.3 Source nodes algorithm to implement the newly determined packet generation rates. Algorithm 3 collects the different $z_{j,n}^k$ values every A seconds, from the different relay nodes, and selects the minimum in order to apply the transmission rates demanded by the most stringent (typically the most saturated) relay node. In some cases, the application demands may be lower than the amount allowed by the relay nodes. Should it be the case, the new transmission rate would simply be the application one.

Algorithm 3: Run at each source node j to implement the newly determined packet generation rate.

while *true* **do**
  initialize all $z_{j,n}^k = \infty$
  collect values $z_{j,n}^k$ from the relay nodes
  compute $s_j^k = min(z_{j,n}^k)$
  for k *in* $\{1, ..., K\}$ **do**
    if $y_j^k < s_j^k$ *and* $x_j^k \geq y_j^k$ **then**
      increase y_j^k to its new value s_j^k
    if $y_j^k > s_j^k$ **then**
      decrease y_j^k to its new value s_j^k
  wait A seconds

4 NUMERICAL RESULTS

We now describe our experimental setup to evaluate the performance of our proposed solution. The network performance parameters of interest are the following:

- **Channel utilization factor**, measured by each node, refers to the occupation of the radio channel averaged over a period of time of 1 second.
- **Packet Delivery Ratio (PDR)** defines the ratio of successfully received packets.
- **Network attained throughput** represents the number of bytes successfully transmitted per second. Note that its value is typically observed together with the PDR.
- **Network transit time** is the time needed by packets to reach their destination through the relay nodes.
- **Jain's index** is a common metric to assess the level of fairness between N entities. Its value ranges from $1/N$ (worst case) to 1 (best case). In our case, it is maximum when all sources receive the same allocation.

4.1 Simulation details

We implemented our solution in C++ within the ns-3 [3] simulator. The wireless communications between nodes are operated using the recent amendment 802.11ac of the IEEE 802.11 standard on a single 20MHz channel with a MCS (Modulation Coding Scheme) set to 0 so that transmissions occur at a physical rate of 6.5 Mbps. Note that all nodes can store in their buffer up to 100 packets.

Source nodes deliver four different classes of traffic, typically corresponding to applications with different traffic properties and QoS requirements. The different classes are presented in Table 2. Without loss of generality and for the sake of clarity, we let the lengths of packets be constant (200 Bytes) and the packet generation rate at source nodes be equal to 100 packets per second (specifically, the time between two successive packet generations is exponentially distributed with a mean of 10 ms). Note that class priorities are ranked in an ascending manner so that class 1 is viewed as the most critical. Nonetheless, let us recall that in our context, we consider that the traffic from all class matters. It follows that our goal is not to entirely guarantee the best possible performance for class 1 traffic at the expense of the other classes but rather to be able to favor class 1 while keeping the communications from lower classes alive. Note that all communications are performed using the transport protocol UDP.

Paths between source and destination nodes are found using the AODV [13] routing protocol wherein the weight of links derives from the expected number of frames transmitted to send a packet (ETX). It is important to mention that the ns-3 simulator includes by default the AODV routing protocol and the hop count metric. In addition, the ETX metric implemented within the AODV by [8] was used in our experiments. Finally, as it was mentioned in previous sections, our solution works independently to the routing protocol. Therefore, other routing protocols can be used without problem.

All simulations are run for a period of 400 sec so as to ensure the discovery of the steady-state regime, where for each scenario, different simulation runs were performed. Each run was configured with different random seeds in order to obtain the network performance parameters explained before. Table 3 details the main simulation parameters used, and Table 4 presents the values used for our distributed solution.

4.2 Scenario A: Tree topology

In our first scenario, we consider a tree topology made of six nodes with three sources, two relays and one destination as depicted previously in Figure 2. As discussed above, each source node (S1, S2 and S3) delivers traffic from four different classes. In any case, packets are first sent to node R1, then forwarded to node node R2 before ultimately reaching node D. To ensure the network topologies shown in Figures 1 and 2, 80 meters was chosen as the distance between nodes located on the sides. On the other hand, 113 meters was chosen as the distance between nodes located on the opposite side. Therefore, with these selected distances and the default propagation model values, each node only establishes

Table 2. NAN applications transmitted over the Smart Grid.

Class	Applications	Packet lengths (Bytes)	Packet generation rate (per sec)
1	Demand Response, Outage Management	200 Deterministic	100 Exponential
2	Video surveillance, Overhead Transmission Line Monitoring, Substation Automation systems (SASs)	200 Deterministic	100 Exponential
3	Home Energy Managament (HEM), Electric Vehicles (EVs) Charging	200 Deterministic	100 Exponential
4	Meter Data Management	200 Deterministic	100 Exponential

Table 3. Main simulation parameters.

Description	Value
Network simulator.	ns-3.28
Simulation time	400 s
Tranport layer	UDP
Random number generator	MRG32k3a
Routing protocol	AODV
Routing metric	ETX
Queue size	100 packets
Wireless physical layer	802.11ac
Modulation coding scheme	0
Short guard interval	1
Frame aggregation factor	0
Channel width	20 MHz

Table 4. Distributed congestion control parameters.

Parameter	Value
K	4
A	1 sec
D	75 %
I	5 %
$[\alpha_1, \alpha_2, \alpha_3, \alpha_4]$	[0.9, 0.8, 0.7, 0.5]
$[\delta_1, \delta_2, \delta_3, \delta_4]$	[1, 0.6, 0.4, 0.2]

link with its neighbors. Note that with a packet generation rate at each source node of 400 pkt/sec (viz 0.64 Mbps), the combined workload of S1, S2 and S3 amounts to 1.92 Mbps and will undoubtedly overflow node R1 whose transmission rate of 6.5 Mbps is reduced by half due to DCF overheads and shared among the 4 nodes neighboring R1.

We now compare the performance attained by the NAN applications with and without our proposed solution. To begin with, we look at the channel utilization factor at the relay node R1. Channel utilization here refers to the occupation factor of the radio channel averaged over a period of time of 1 second. Figure 3 shows the corresponding results. The blue curve (labelled *not controlled*) represents the case where our solution does not operate and sources are free to generate packets at their initial rate, namely 100 pkt/sec for each class. The channel utilization factor nears 0.81 (meaning that R1 perceives the radio channel as busy about 80% of the time). In practice, the radio channel can be viewed as fully saturated since the 20% left corresponds to unavoidable delays such as those incurred by the contention window. Unlike the blue curve, the green curve corresponding to the case wherein our solution is applied (labelled *controlled*). The ρ factor oscillates between 0.6 and 0.8 with a mean value of 0.71. This behavior indicates that, with our solution, short saturations may occur but they are rapidly vanishing.

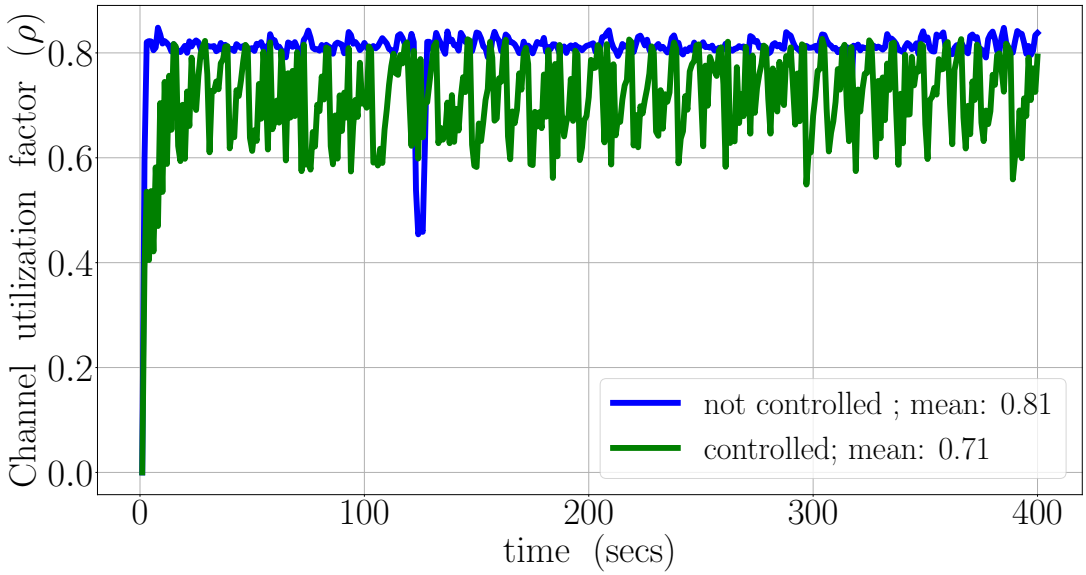

Fig. 3. Scenario A - Evolution of the channel utilization factor at R1 node.

We now study the time evolution of the packet generation rates at the source node S1. Figure 4 represents the corresponding results for each of the four classes. As expected, in the absence of our solution, the packet generation rate holds its initial value of 100 pkt/sec for any class. On the other hand, with the help of our solution, we can observe that the classes exhibit different patterns. For instance, the boxplot for class 1 indicates that the generation rate may vary from less than 95 up to 100 pkt/sec while its median value is 100 pkt/sec. Looking at the less critical classes, we note that the median value for their packet generation rate tends to be significantly smaller but yet far from 0 though.

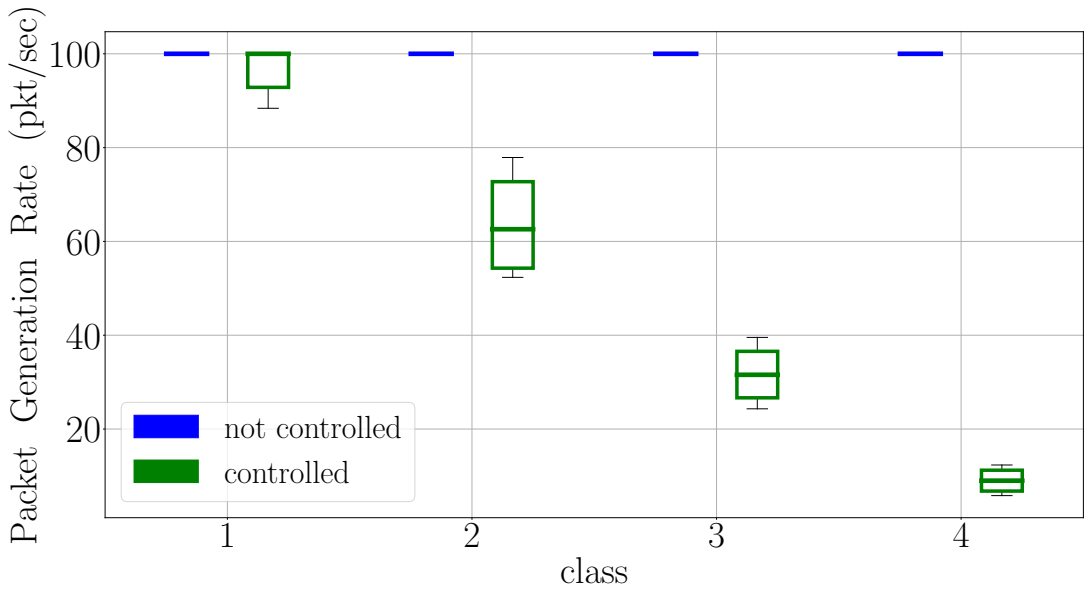

Fig. 4. Scenario 1 - Packet generation rate at node S1 for each class.

In Figure 5, we study the packet delivery ratio (PDR) attained for each class with and without our solution. Because the radio channel tends to be saturated in the absence of our solution (see Figure 3), the corresponding PDR only peaks at 55%. When our solution is applied, network sources are regulated so that congestions do not last thereby preventing packet losses. It follows that the the PDR attains 100%.

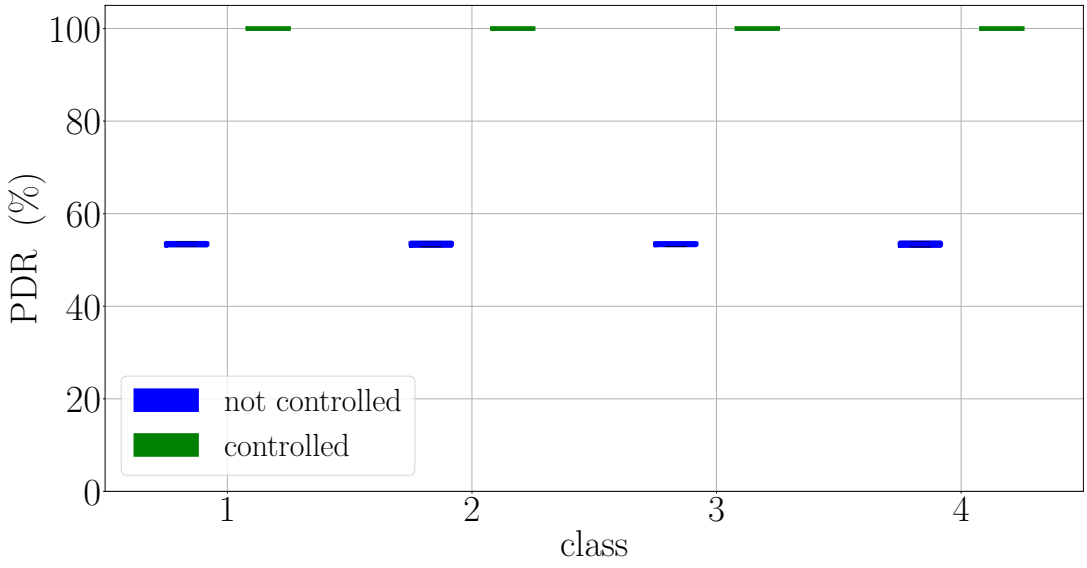

Fig. 5. Scenario A - Packet delivery ratio at node D for each class.

Figure 6 shows the throughout attained by the different applications depending on the class they belong to. Note that here the attained throughput refers to the number of bytes that were successfully transmitted from the sources to the destination node. Not surprisingly, we notice that in the absence of our solution, all classes exhibit the same pattern with a constant throughput equal to 0.28 Mbps. However, when using our solution, each class obtains a level of throughput according to its degree of criticality. Note that the attained throughputs in Figure 6 can be derived as the product of the packet generation rates shown in Figure 4 and the PDR from Figure 5 (converting packets into Mbits).

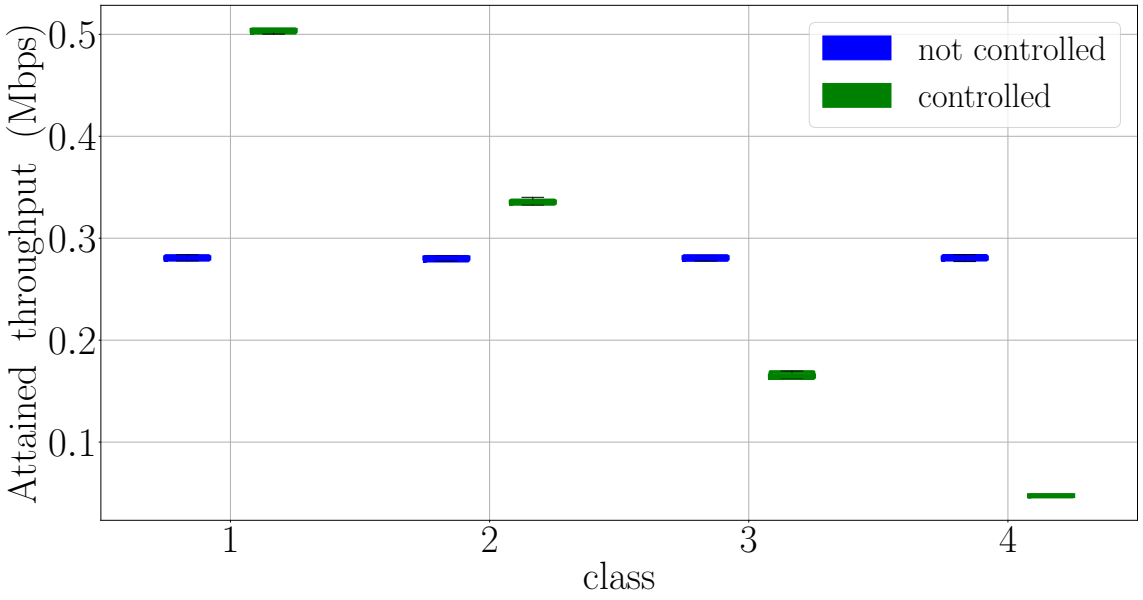

Fig. 6. Scenario A - Attained throughput at node D for each class.

Overall, this first scenario illustrates the usefulness of our solution. In the next section, we consider a second scenario where the number of nodes (sources and relays) is much larger.

4.3 Scenario B: Smart grid scenario

Our second scenario deals with a set of 16 nodes where all nodes but the destination (Data concentrator) are both sources and relays. Nodes are arranged, grid-like, into 4 rows and 4 columns as illustrated previously by Figure 1. Clearly, given the workload delivered by each node, the network will sustain congestion unless some form of control is made on the source packet generation rates.

First, we study the packet delivery ratio attained by nodes for each of their class. Figure 7 indicates that in the absence of our solution roughly 18% packets will ultimately reaching their destination. Said differently, most of them will be lost on their way towards the data concentrator. When using our solution, Figure 7 shows that virtually all generated packets successfully reach the destination node.

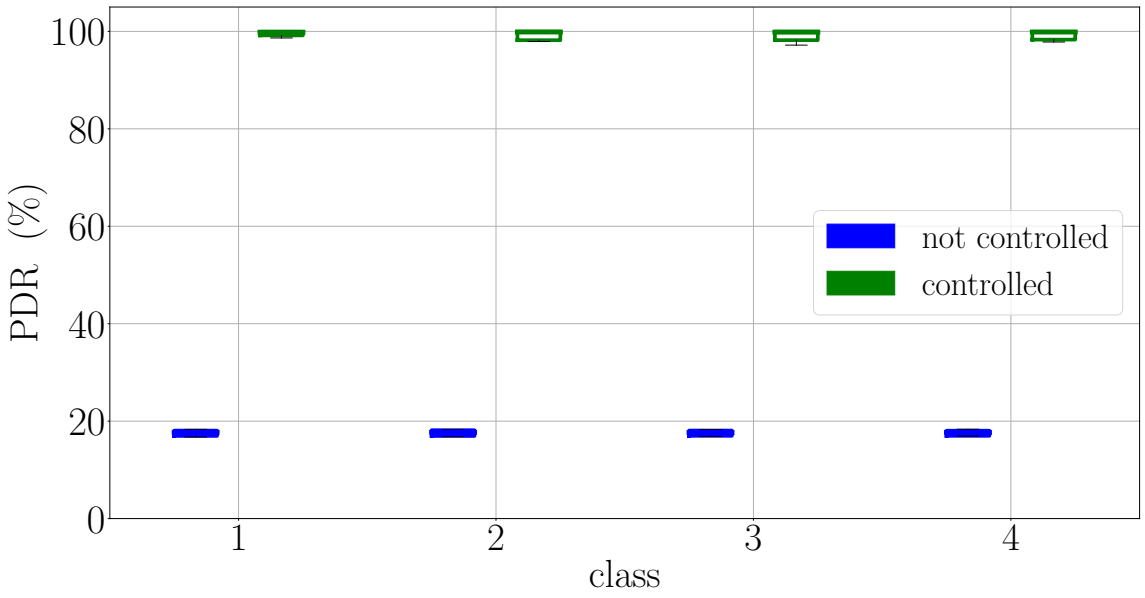

Fig. 7. Scenario B - Packet delivery ratio at node D for each class.

To have a better understanding on the network performance, we look at the attained throughput. Figure 8 shows that, without our solution, each class manages to get an attained throughput of about 0.47 Mbps. On the other hand, when our solution is applied, the attained throughput strongly varies with the class the traffic originates from. Nonetheless, all classes manage to get an access, be it small, to the network communication resources.

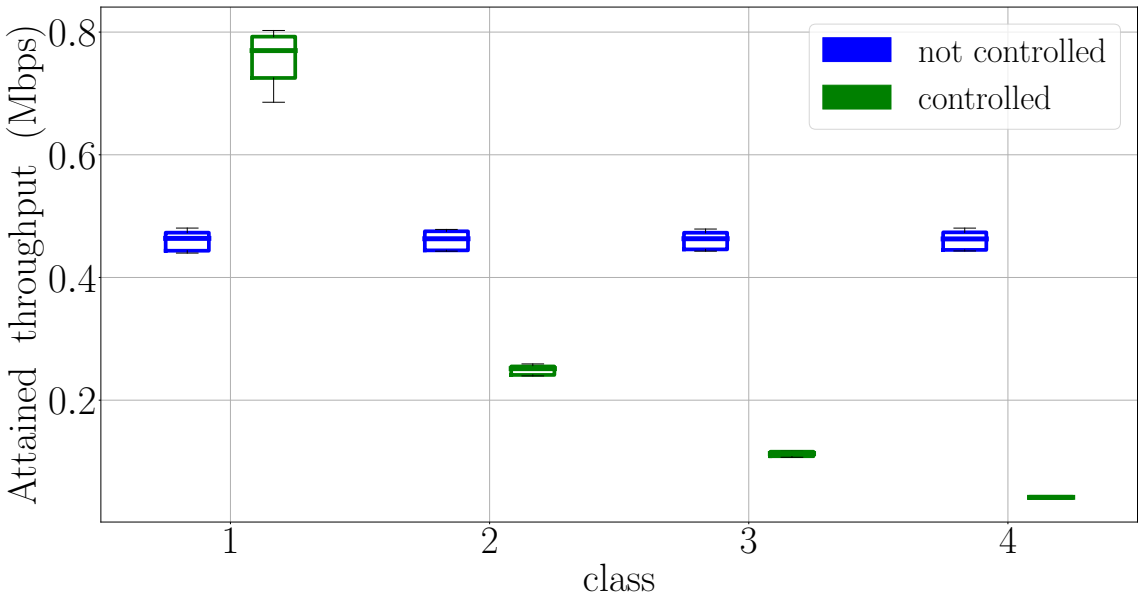

Fig. 8. Scenario B - Attained throughput at node D for each class.

So far, the combination of Figures 7 and 8 has shown that the implementation of our solution enables all applications to experience a near 100% packet delivery ratio and to differentiate their attained throughput based on their assumed criticality while maintaining a high level of the network resource utilization. We now investigate if nodes experience different levels of attained throughput based on their location. For instance, one can expect that nodes further away from the data concentrator undergo poorer performance than nodes in the vicinity of it. To do that, we compute a measure of fairness, namely the Jain's index, which somehow reflects the variability in the attained throughput by nodes. Figure 9 represents the corresponding results. While the Jain's index attains a value of 0.3 in the absence of our solution, its value ranges from 0.8 to 0.95 with the help of our solution. These results indicate that the 15 source nodes and their different classes of traffic of the smart grid topology are much more fair in their sharing of the network resources when our solution is applied.

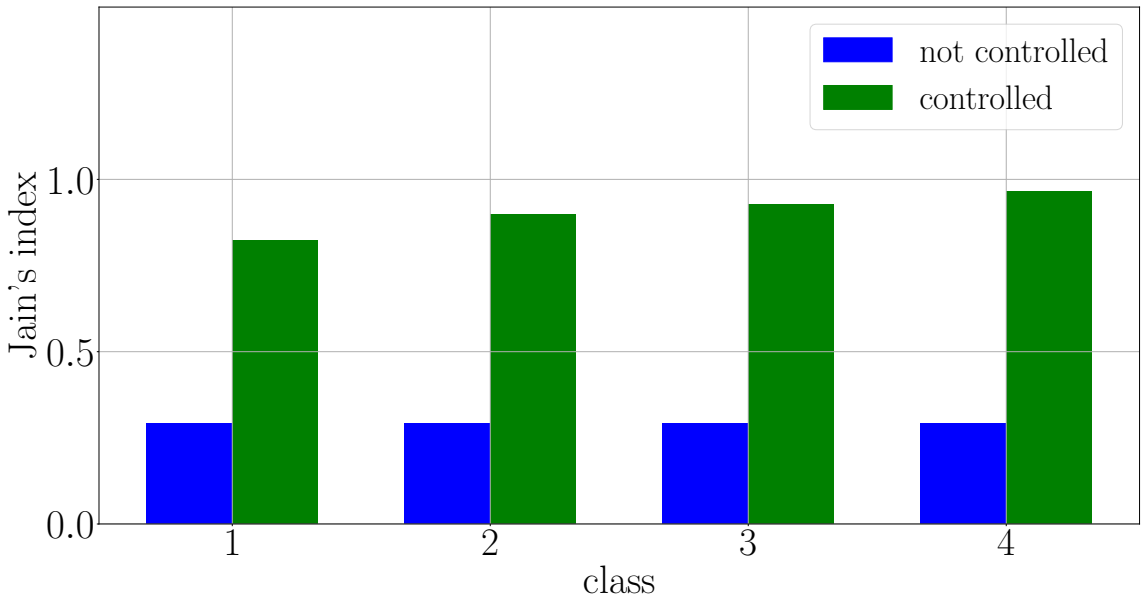

Fig. 9. Scenario B - Fairness on the attained throughputs.

We now investigate the network transit time experienced by packets sent from the nodes. Figure 10 represents the network transit time found for each traffic class with and without our solution. We observe that without our solution the network transit times tend to vary between 443 to 4307 ms. On the other hand, using our solution enables to decrease these transit times to a range of 1 to 14 ms. In practice, our proposed solution throttles the sources at lower paces so that buffers at the relay nodes are less occupied and queueing delays significantly reduced.

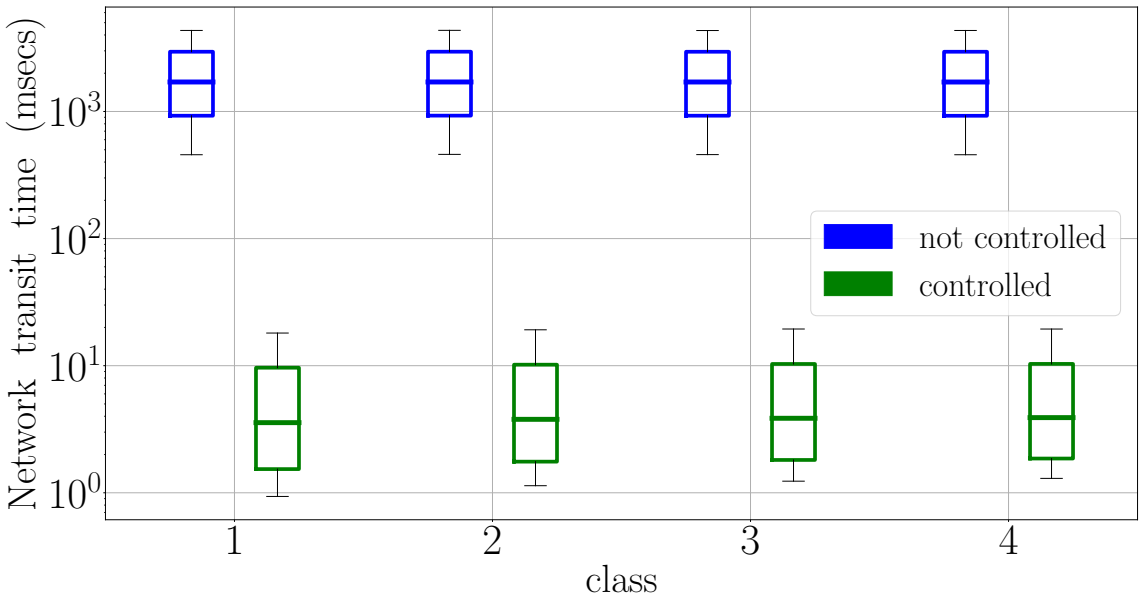

Fig. 10. Scenario B - Network transit time.

Finally, we study the overhead due to the additional signaling traffic caused by our solution. While the signaling traffic represents between 3 and 4% of the total network load, this traffic enables to significantly decrease the packet generation rate at the source nodes and therefore has a positive impact on the total network load.

5 CONCLUSIONS AND FUTURE WORK

In this work, we propose and evaluate the performance of a distributed congestion control mechanism for the Smart Grid Neighborhood Area Networks. The scenario considered is built by intelligent devices interconnected with each other through an IEEE 802.11 ad hoc network that uses the IEEE 802.11ac standard as the physical layer. The proposed solution offers traffic differentiation (depending on the application criticality) and fair distribution of network resources between all the network nodes.

In order to assess the network performance, two scenarios have been tested with the help of a discrete-event simulator. Numerical results show the improvements obtained from our congestion control mechanism in terms of packet delivery ratio, network throughput, transit time as well as fairness between different traffic sources.

As future lines of work, mathematical formulation will be made in order to forecast how the throughput will be distributed between different classes. Besides, different scenarios will be implemented. That is, the network performance of our solution will be evaluated, when the grid size is increased from 16 to 36 nodes. On the other hand, the traffic from the control center towards downstream to the the smart meters, will be implemented and evaluated with our solution. Finally, the different variables will be adjusted in order to provide the minimum QoS requirements for all classes.

ACKNOWLEDGMENTS

This work was supported by the Spanish Research Council under projects INRISCO (TEC2014-54335-C4-1-R) and MAGOS (TEC2017-84197-C4-3-R), and Juan Pablo Astudillo León is the recipient of a full scholarship from the Secretaria Nacional de Educación Superior, Ciencia y Tecnología.

REFERENCES

[1] Juan Pablo Astudillo León and Luis J. De la Cruz Llopis. 2018. A Joint Multi-Path and Multi-Channel Protocol for Traffic Routing in Smart Grid Neighborhood Area Networks. *Sensors* 18, 11 (2018). https://doi.org/10.3390/s18114052

[2] T Clausen and P Jacquet. 2003. Optimized Link State Routing Protocol (OLSR). *Internet Engineering Task Force (IETF)* 4 (2003), 75. https://doi.org/10.1.1.11.620 arXiv:arXiv:1011.1669v3

[3] NS-3 Consortium. 2019. Ns-3 a Discrete-Event Network Simulator. https://www.nsnam.org/

[4] Xiaoheng Deng, Lifang He, Xu Li, Qiang Liu, Lin Cai, and Zhigang Chen. 2016. A reliable QoS-aware routing scheme for neighbor area network in smart grid. *Peer-to-Peer Networking and Applications* 9, 4 (jul 2016), 616–627. https://doi.org/10.1007/s12083-015-0331-5

[5] IEEE Standard for Information technology Telecommunications, information exchange between systems Local, metropolitan area networks Specific requirements Part 11: Wireless LAN Medium Access Control (MAC), and Physical Layer (PHY) Specifications. 2016. IEEE Std 802.11-2016 (Revision of IEEE Std 802.11-2012), pp. 1 – 3534. *IEEE Computer Society* (2016).

[6] Hamid Gharavi and Bin Hu. 2011. Multigate Communication Network for Smart Grid. *Proc. IEEE* 99, 6 (jun 2011), 1028–1045. https://doi.org/10.1109/JPROC.2011.2123851

[7] V. Cagri Gungor, Dilan Sahin, Taskin Kocak, Salih Ergut, Concettina Buccella, Carlo Cecati, and Gerhard P. Hancke. 2013. A Survey on Smart Grid Potential Applications and Communication Requirements. *IEEE Transactions on Industrial Informatics* 9, 1 (feb 2013), 28–42. https://doi.org/10.1109/TII.2012.2218253 arXiv:arXiv:1011.1669v3

[8] Nenad J Jevtić and Marija Z Malnar. 2018. Implementation of ETX Metric within the AODV Protocol in the NS-3 Simulator. *Telfor Journal* 10, 1 (2018), 20–25.

[9] Ji-Sun Jung, Keun-Woo Lim, Jae-Beom Kim, Young-Bae Ko, Younghyun Kim, and Sang-Yeom Lee. 2011. Improving IEEE 802.11s Wireless Mesh Networks for Reliable Routing in the Smart Grid Infrastructure. In *2011 IEEE International Conference on Communications Workshops (ICC)*. IEEE, 1–5. https://doi.org/10.1109/iccw.2011.5963578

[10] Jaebeom Kim, Dabin Kim, Keun-Woo Lim, Young-Bae Ko, and Sang-Youm Lee. 2012. Improving the reliability of IEEE 802.11s based wireless mesh networks for smart grid systems. *Journal of Communications and Networks* 14, 6 (dec 2012), 629–639. https://doi.org/10.1109/JCN.2012.00029

[11] Juan Pablo Astudillo León and Luis J. de la Cruz Llopis. 2019. Emergency aware congestion control for smart grid neighborhood area networks. *Ad Hoc Networks* 93 (2019), 101898. https://doi.org/10.1016/j.adhoc.2019.101898

[12] Weixiao Meng, Ruofei Ma, and Hsiao Hwa Chen. 2014. Smart grid neighborhood area networks: A survey. *IEEE Network* 28, 1 (jan 2014), 24–32. https://doi.org/10.1109/MNET.2014.6724103

[13] C Perkins, E Belding-Royer, and S Das. 2003. *Ad hoc On-Demand Distance Vector (AODV) Routing, RFC 3561 (Experimental).* Technical Report 3561. http://www.ietf.org/rfc/rfc3561.txt

[14] N Shaukat, SM Ali, CA Mehmood, B Khan, M Jawad, U Farid, Z Ullah, SM Anwar, and M Majid. 2017. A survey on consumers empowerment, communication technologies, and renewable generation penetration within Smart Grid. *Renewable and Sustainable Energy Reviews* (2017).

[15] Yakubu Tsado, Kelum Gamage, Bamidele Adebisi, David Lund, Khaled Rabie, and Augustine Ikpehai. 2017. Improving the Reliability of Optimised Link State Routing in a Smart Grid Neighbour Area Network based Wireless Mesh Network Using Multiple Metrics. *Energies* 10, 12 (feb 2017), 287. https://doi.org/10.3390/en10030287

[16] Yakubu Tsado, Kelum A. A. Gamage, David Lund, and Bamidele Adebisi. 2016. Performance analysis of variable Smart Grid traffic over ad hoc Wireless Mesh Networks. In *2016 International Conference on Smart Systems and Technologies (SST)*. IEEE, 81–86. https://doi.org/10.1109/SST.2016.7765637